Scene Text Understanding

Sanjeev Satheesh SSANJEEV@STANFORD.EDU
Department of Computer Science, Stanford University

Abstract

The task is character recognition from real world images such as sign boards. This report presents a work in progrss along with the some interesting results. Sparse autoencoders have been used to learn a codebook of basis functions, and the settings for obtaining the best results on this task is studied

1. Introduction

While optical character recognition of scanned documents is considered a solved problem, with nearly 99% accuracy; detecting and reading text in natural scenes is still a very open problem. This is due to various reasons such as non uniformity of text fonts seen, skew from the photograph, and improper lighting, glare, illumination and other such factors. The accuracy of text detection and recognition from natural scenes is around 70%. This problem is known in literature as Scene Text Understanding or more informally as PhotoOCR. In this project the problem of scene text recognition is undertaken. Given a bounding box around each of the characters in a photo of a natural scene the task is to recognize what the character within the bounding box is. This report presents a work in progress along with the results obtained so far.

2. Datasets

The following datasets have been assembled:

- ICDAR 2003/2005 dataset, a set of 480 images of real world signs used in the ICDAR Robust Reading competition (Lucas, 2006)
- Microsoft's Scene text understanding dataset composed of 305 real world signs, used in (Epshtein et al., 2010)
- Dataset composed of 100 sign boards used for scene text "reading" used in (Weinman et al., 2005)
- Synthetic data, for learning the general features of all characters. This includes 3 sets of 10000 images for each character. Each image is 32 by 32 with a single character on it. The easy set has the character centered on the image without any distortions. The characters in the medium dataset are distorted, tilted and/or translated towards a corner, but on a plain background, and those in the hard dataset are distorted and on background that is obtained from a realworld photographs.

The first three datasets provide bounding boxes for each character in the image, along with a label but together have only 1000 images. All the datasets have been homogenized into a single format for easier access.

3. Feature Learning

The features to use for classification, especially in the case of Image Processing tasks, is often a result of weeks and sometimes months of research and is usually specialized for that application. One way to automate the process of choosing features is feature learning. The stacked sparse autoencoder method presented in (Papusha et al.) provides a method for learning the bases that a dataset is made of. Each class of data in any dataset is made of different combination of these bases, and noting which combination of features reconstruct the data item usually forms a good feature vector for classification. The implementation used in this project follows closely the one in (Papusha et al.), and the loss includes reconstruction error, sparcity penalty using cross entropy error and regularization.

To learn the bases, the sparse autoencoder requires a large number of patches. The patches were obtained from the easy dataset by taking 4 8 by 8 patches from each of the samples. The Autoencoder was then

trained on these patches with a target activation of 0.03 and regularization constant at 0.003 for exactly 750 iterations of batch gradient descent.

3.1. Whitening

The whitening transform is recently found to be very effective in improving the performance of autoencoders. We tried two different versions of whitening transforms.

3.2. ZCA

Nearby pixels in images are correlated, and if left unaltered, the Autoencoder will learn second order relationships like this correlation, which is not of practical purpose for this task. The ZCA whitening transform converts the dataset so that the covariance matrix of the pixels of the resulting transformed dataset is a diagonal matrix (Krizhevsky & Hinton, 2009). The diagonalization is achieved by calculating the PCA of the data matrix. Because the image statistics are roughly stationary, the eigenvectors of the covariance matrix will essentially be equivalent to the Fourier bases.

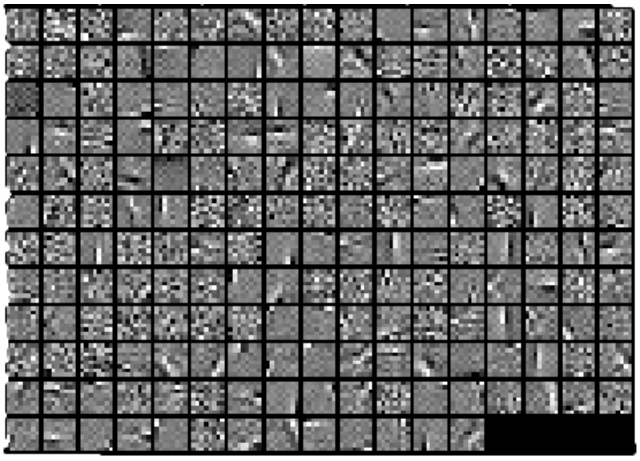

Figure 1. 200 bases learnt after ZCA whitening of the images.

3.3. Combined whitening/low pass filter

The seminal work on Autoencoders, (Olshausen & Field, 1997) use a similar whitening transform, but operate directly on the frequency spectrum of the images. Natural images have $1/f^2$ power spectrum which results in large inequities in variance along different bases with low frequencies having high variance. To remove this skew we "sphere" the data by equalizing the variance in all directions (Friedman, 1987). The combined whitening/low-pass filter used to preprocess the data had a frequency response of

$R(f) = fe^{-(f/f_o)^4}$

where the first component is a whitening filter, which attenuates the lower frequncies and amplifies the higher frequencies; and the second component is a low pass filter which cuts out the power at the highest spatial frequencies.

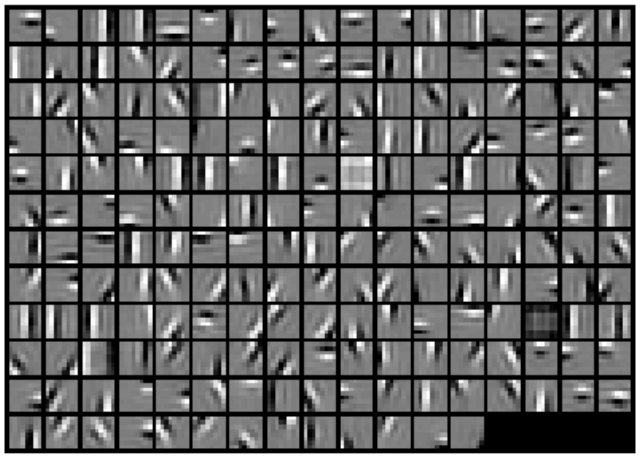

Figure 2. 200 bases learnt after applying the combined whitening/low pass filtering on images.

3.4. Binarization

An experiment was attempted to learn bases from the binarized data items. For binarization, we used the thresholding method listed in (Wayne, 1986)

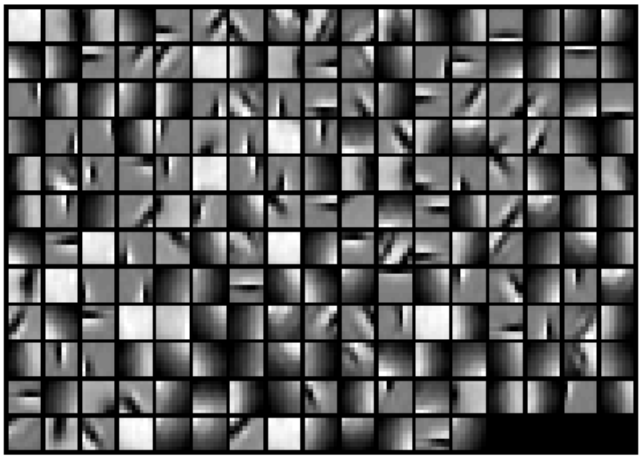

Figure 3. 200 bases learnt from the Binarized data.

All three sets of bases though different visually gave almost similar results in the classification task. Results mentioned in the results section all use bases from figure 2

The effect of the three transforms on a data item in the easy dataset is visualized in the figure 4. ZCA, though much popular in general did not give as clear

whitening transforms or better bases. It is clear that the second method provides much clearer bases than others.

Figure 4. The first column shows the original image, the second one is the result of ZCA, the third is the result of the combined whitening/low pass filter and the last column shows the binarized image

4. Feature Extraction

For extracting the features from a sample data item, each of the learnt bases is densely convolved on the data item, and $sigmoid(wx + b)$ is calculated which is the same as the activation of the hidden layer of the sparse autoencoder. The activations are then pooled into 4 or 9 regions using max pooling and sum pooling. Max pooling gave better results than sum pooling in the classification task, as seen in figure 6

5. Classification

For multinomial classification softmax regression with L2 regularization was used. SVM with L2 regularization was also tried but it did not offer any significant improvements in time or accuracy.

6. Results

Unless otherwise specified, the following results are experiments run on the the upper case characters of the easy dataset in grayscale.

6.1. Type of pooling

Figure 5 shows the effect of different types of pooling. The results are on the upper case characters of the easy dataset.

6.2. Count of pooling regions

Increasing the number of pooling regions from 4 to 9 gave marginal but consistent improvement in test accuracies. Figure 6 demonstrates this behaviour.

6.3. Number of bases used

Increasing the number of bases learnt did not improve the results as the bases starting duplicating.

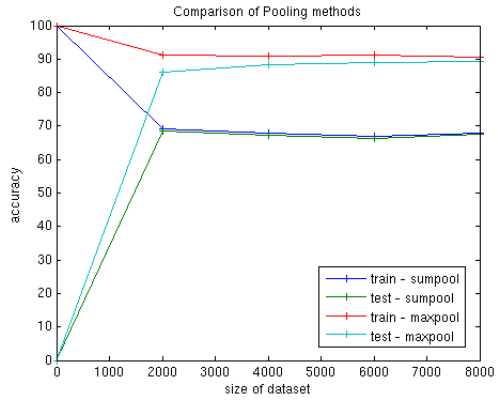

Figure 5. Plot showing performance of feature extraction using sum pooling and max pooling against the number of images per class in the upper case easy dataset

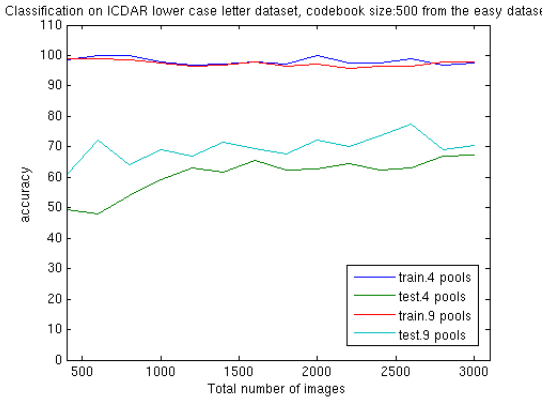

Figure 6. Plot showing the performance increase obtained from increasaing the number of pooling regions against the total number of images from all classes. Classification was done using non regularized softmax regression

6.4. Classification results

As the previous plots indicate the best results on easy dataset was using max-pooling which topped at 91%.This is not the ideal result as commercial OCR engines can be expected to perform much better and have accuracies around 99%. On the hard dataset, the best results were at 49% test accuracy on the full set of 26,000 images of upper case characters. On the ICDAR dataset the best results were around 73% for both the lower case and upper case characters. The learning curves has not converged even though the full set of 3000 character samples were exhausted. The results are better than the preliminary results obtained using HOG features on all the datasets.

7. Future work

These results could be improved by small percentages still more. Among the things to try are

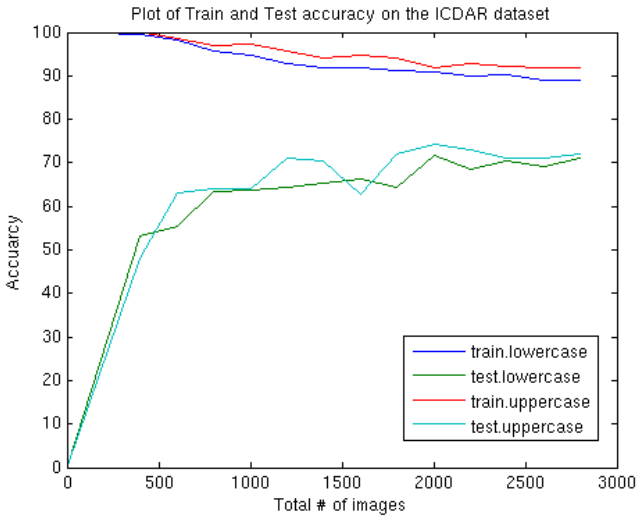

Figure 7. Plot showing the performance on the ICDAR data. The features used are the 500 bases learnt from the combined filter whitening transform

- Move to color. With 3 times more data in the feature vectors, the accuracies may improve by a few points
- Sparse autoencoders seem to be able to learn bases that are more or less edge detecting filters. Use other methods such as K Means to learn bases which could be more diverse.
- Learn bases better suited for this classification task by using tuning methods
- Test on the other real world datasets

In the tests on the ICDAR data, we tested on upper and lower cases separately, and training and testing were both on the ICDAR data. In order to build a robust system we need to use the real images only for testing. Testing for accuracy on this setup are being conducted. Also in the pipleline is a full 62 class (upper+lower+digits)classification test.

Acknowledgments

This is joint work with Blake Carpenter, Carl Case, Bipin Suresh, Tao Wang and advised by Adam Coates. The generation of the synthetic datasets was done by Bipin Suresh. The experiments using HOG provided in the results section was provided by Bipin Suresh and Tao Wang.

References

Epshtein, B., Ofek, E., and Wexler, Y. Detecting text in natural scenes with stroke width transform. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2963–2970. IEEE, 2010.

Friedman, J.H. Exploratory projection pursuit. *Journal of the American statistical association*, 82(397): 249–266, 1987. ISSN 0162-1459.

Krizhevsky, A. and Hinton, GE. Learning multiple layers of features from tiny images. *Master's Thesis, Department of Computer Science, University of Toronto*, 2009.

Lucas, S.M. ICDAR 2005 text locating competition results. In *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*, pp. 80–84. IEEE, 2006. ISBN 0769524206.

Olshausen, B.A. and Field, D.J. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research*, 37(23):3311–3325, 1997. ISSN 0042-6989.

Papusha, I., Ngiam, J., and Ng, A. Stacked Autoencoders for Semi–Supervised Learning.

Wayne, N. *An Introduction to Digital image processing*. Prentice-Hall International London, 1986.

Weinman, J., Hanson, A., and McCallum, A. Sign detection in natural images with conditional random fields. In *Machine Learning for Signal Processing, 2004. Proceedings of the 2004 14th IEEE Signal Processing Society Workshop*, pp. 549–558. IEEE, 2005. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=142